
CS/ST 480

HOMEWORK 4 (DUE MARCH 24TH)

1. Dishonest Casino [15 pts]

Use the Hidden Markov Model (HMM) for the “dishonest casino” in the class handout (also Durbin et. al. chapter 3) to answer the following questions.

- What is the probability of obtaining the sequence **1465466636** ? Compute this probability using the forward algorithm and then again using the backward algorithm.
- For each position in the sequence, calculate the probability that the FAIR die was used to obtain the outcome at that position. Use POSTERIOR DECODING to estimate the hidden states (FAIR or LOADED) corresponding to the observed sequence.
- Compute the most probable sequence of states that generates the observed sequence given above. You should fill up a 2-by-10 dynamic programming table.

2. 2-State HMM for predicting G+C rich regions [40 pts]

Write a program that implements a 2-state HMM for detecting G+C rich regions in the *Pyrococcus furiosus* DSM 3638 (GenBank: AE009950, Refseq: NC_003413) genome sequence. This sequence may be downloaded from

http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi??db=nucleotide&val=NC_003413

Conceptually, state 1 of your HMM will correspond to the more frequent ‘A+T rich’ state, whereas state 2 will correspond to the less frequent ‘G+C-rich’ state. Specific details are as follows:

- The starting parameter values for the HMM (*see Klein et al (2002), PNAS 99: 7542-47 for some background information*) should be as follows:
 - Transition probabilities a_{ij} are $a_{11} = 0.999$, $a_{12} = 0.001$, $a_{21} = 0.01$, $a_{22} = 0.99$.
 - Initial state probabilities (i.e. the transition probabilities from the ‘begin’ state into state 1 or 2) should be $a_{01} = 0.996$ for state 1, and $a_{02} = 0.004$ for state 2; *hold these fixed throughout the Viterbi training in part (b)*.
 - Emission probabilities (*which should also be held fixed for the Viterbi training in part (b)*) are $e_A = e_T = 0.296$, $e_G = e_C = 0.204$ for state 1; $e_A = e_T = 0.159$, $e_G = e_C = 0.341$ for state 2.
- Use Viterbi training to find improved parameter estimates for the transition probabilities, holding the emission and initiation probabilities fixed at the values specified above. Run the training for 10 iterations, where for each iteration
 - use dynamic programming to find the highest probability underlying state sequence

- use this state sequence to compute the number of states of each of the two types (1 and 2), and the number of segments of each type (where a segment consists of a contiguous series of states of the same type, that is preceded and followed by states of the opposite type or the beginning or end of the sequence), and
- new transition probabilities to be used in the next iteration.

(c) Your output should provide

- the information described above in part (b) – i.e. numbers of states and segments, and new probability values), for each of the 10 iterations. (Give probabilities to 3 decimal places only.)
- the list of ‘G+C-rich’ segments (corresponding to the segments having state 2 as the underlying state) after the final (10th) round of Viterbi training.
- a description of the first 10 of the ‘G+C-rich’ segments you found in terms of how they relate to Genbank annotations.

3. [Baum-Welch training of HMM (30 PTS)]

Using the same HMM and dataset as in the previous problem, write a program that implements EM (Baum-Welch) training instead of Viterbi training. Use the same starting parameter values, but in contrast to the previous problem, you should not hold any parameter values fixed – allow all of them to change with each iteration. Compute the log-likelihood (use base 2) of the sequence at each iteration, and run the program until the increase in log-likelihood between successive iterations becomes less than 0.1. You should check that the log-likelihood increases with each iteration – if it doesn’t, something is wrong with your program.

Your output should provide

- the number of iterations until convergence
- the final log-likelihood
- the final emission and transition probabilities

4. Gene prediction using a simple HMM [50 PTS]

Write a program that implements an HMM for predicting genes in the genome sequence of *Escherichia coli K12*. The Ecoli K12 sequence may be downloaded from

http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi??db=nucleotide&val=NC_000913

Specific details are as follows:

- (a) The HMM has 4 states, numbered 1 through 4 (not counting the BEGIN state 0).
- (b) State 0 is the BEGIN state. The markov chain always starts at the BEGIN state.
- (c) State 1 corresponds to the intergenic region. It emits single nucleotides (A, C, G, or T). The emission probabilities are chosen to be 0.246, 0.254, 0.254, 0.246 for A, C, G, T. These probabilities roughly correspond to the relative frequencies of occurrence of A, C, G, T, in E.coli K12.

- (d) State 2 corresponds to a start codon. It emits one of three codons – ATG, GTG, or TTG, each with appropriate probabilities. Based on observed frequencies in E. coli K12 these probabilities are taken to be $P(ATG) = 0.918770$, $P(GTG) = 0.069155$, $P(TTG) = 0.012075$.
- (c) State 3 corresponds to an internal codon of a gene. It emits any one of 61 codons that does not correspond to a stop codon (i.e. the stop codons TAA, TGA, TAG have emission probabilities 0). The codon emission probabilities for these 61 codons are chosen according to codon usage statistics for E.coli K12 in the known coding regions. The probabilities are given in the table below.

	Axx	Cxx	Gxx	Txx
xAA	0.0354810	0.0134260	0.0426069	0.0000000
xCA	0.0051414	0.0080348	0.0197384	0.0057318
xGA	0.0012651	0.0029648	0.0062151	0.0000000
xTA	0.0028253	0.0031692	0.0113759	0.0109672
xAC	0.0240623	0.0110839	0.0219992	0.0135654
xCC	0.0252626	0.0043791	0.0247629	0.0092674
xGC	0.0156805	0.0239844	0.0322170	0.0066173
xTC	0.0269137	0.0103347	0.0150154	0.0181521
xAG	0.0107207	0.0301054	0.0184895	0.0000000
xCG	0.0136985	0.0264109	0.0356684	0.0083852
xGG	0.0008434	0.0044342	0.0099843	0.0141137
xTG	0.0252917	0.0568471	0.0267254	0.0123004
xAT	0.0142305	0.0121285	0.0318117	0.0148370
xCT	0.0092804	0.0062508	0.0163421	0.0090534
xGT	0.0067244	0.0247468	0.0277472	0.0047099
xTT	0.0279892	0.0092609	0.0191319	0.0194821

- (d) State 4 corresponds to a stop codon – TAA, TGA, or TAG. So emission probabilities for the other 61 codons are zero. The emission probabilities for TAA, TGA, TAG are chosen according to their frequencies of occurrence in E.coli K12. These are $P(TAA) = 0.663008$, $P(TGA) = 0.273326$, $P(TAG) = 0.063666$.

- (f) The allowed transitions are as follows:

- State 0 transitions only to state 1 or state 2.
- State 1 transitions only to itself or to state 2.
- State 2 transitions only to state 3.
- State 3 transitions only to itself or to state 4.
- State 4 transitions to state 1 with probability 1.

- (j) Initial guesses for transition probabilities are

$$a_{01} = 1/2, a_{02} = 1/2, a_{11} = 0.998, a_{12} = 0.002, a_{23} = 1, a_{33} = 0.995, a_{34} = 0.005.$$

All a_{ij} not mentioned above are set to zero.

Use Viterbi training to find improved parameter estimates. No parameter is held fixed during the Viterbi training. Run the training for 5 iterations, where for each iteration you:

1. Use dynamic programming to find the highest probability underlying state sequence.
2. Using this state sequence, compute
 - The numbers of top strand and bottom strand genes.
 - New transition probabilities and new emission probabilities, to be used in the next iteration.

Your output should provide

- For each of the 5 iterations, the numbers of top strand and bottom strand genes given by the Viterbi parse, and the new transition and emission probabilities. Give probabilities to 5 decimal places. List all of the probabilities, even when they are 0.
- Compare your results with your gene prediction results from problem 4 of Homework 1.

5. 8-STATE HMM FOR CpG ISLANDS [50 PTS]

In this problem you will use a Hidden Markov Model to identify CpG-islands in a segment consisting of the first 1,000,000 base pairs in the genomic sequence of (*Human Chromosome 22*). Information about CpG islands in human chromosome 22 is available from

<http://www.ncbi.nlm.nih.gov/mapview/maps.cgi?TAXID=9606&CHR=1&MAPS=cpg&BEG=&END=1000K&thmb=on>

The sequence itself may be downloaded from the course website. You can also check out

<http://www.ncbi.nlm.nih.gov/entrez/viewer.fcgi?val=51511751&view=gbwithparts>

The dinucleotide transition probabilities in CpG-islands are different from that in non CpG-islands. Your HMM will have a total of 8 states – a group of 4 states A+, C+, G+, and T+ which emit A, C, G, and T respectively in CpG-islands, and another group of 4 states A-, C-, G-, and T- correspondingly to normal (non CpG) genomic regions. For transition probabilities within each group we will use the transition probability tables given in the class handout (also see page 50 in Durbin et al's book). Now it is your task to design the transition probabilities between the states across groups and also the initial state probabilities.

- (a) Discuss how you will go about designing the transition probabilities between the states across groups? How about initial state probabilities? Justify the choices you make.
- (b) Use the 'sliding window' approach with window width equal to 300 bp to get an informal estimate of the number of CpG islands that may be present in this sequence. Try other window widths as well.
- (c) Use posterior-decoding to estimate the number of CpG islands in this sequence. Also estimate the average length of a putative CpG island and the average length of a non CpG island.
- (d) Under your HMM model, find the most probable path, given the observed sequence, using the Viterbi algorithm. Use this path to annotate the given sequence as CpG island regions and non CpG island regions. How many CpG islands did you find? Compare your results with Genbank annotation available from the NCBI site mentioned above.